Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт 2**

***з дисципліни:*** «Мови та парадигми програмування»

***з розділу***: «Епік 2:Лінійні та розгалужені алгоритми. Умовні оператори. Константи, змінні»

***Виконав:***

студент групи ШІ-13

Юнко Дмитро Богданович

# **Тема роботи:**

Лінійні та розгалужені алгоритми мови C++. Умовні оператори, їхні види , та використання. Лінійні та розгалужені алгоритми мови C++. Константи, змінні, та їхні види

# **Мета роботи:**

# Лінійні алгоритми та розгалужені алгоритми. Опрацювання базових операцій у алгоритмізації процесів. Використання операторів умови, циклу. Написання Програм використовуючи increment and decrement.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Лінійні алгоритми та розгалужені алгоритми.
* Тема №2: Умовні оператори.
* Тема №3: Константи та змінні.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Лінійні алгоритми та розгалужені алгоритми.
  + Джерела Інформації
    - Конспект лекцій з дисципліни «Алгоритмізація та програмування. Частина 1» Л.І. Мочурад
    - Beginning C++ Programming - From Beginner to Beyond
  + Що опрацьовано:
    - Визначення та властивості лінійних та розгалужених алгоритмів
    - Приклади лінійних та розгалужених алгоритмів
    - Практичне застосування лінійних та розгалужених алгоритмів у програмуванні
  + Статус: Ознайомлений
  + Початок опрацювання теми: 24.10.2023
  + Звершення опрацювання теми: 24.10.2023
* Тема №2: Умовні оператори.
  + Джерела Інформації:
    - Конспект лекцій з дисципліни «Алгоритмізація та програмування. Частина 1» Л.І. Мочурад
    - Beginning C++ Programming - From Beginner to Beyond
  + Що опрацьовано:
    - Визначення та властивості умовних операторів
    - Приклади використання умовних операторів у програмуванні
    - Практичне застосування умовних операторів у програмуванні
  + Статус: Ознайомлений
  + Початок опрацювання теми: 25.10.2023
  + Звершення опрацювання теми: 25.10.2023
* Тема №3: Константи та змінні.
  + Джерела Інформації:
    - Конспект лекцій з дисципліни «Алгоритмізація та програмування. Частина 1» Л.І. Мочурад
    - Beginning C++ Programming - From Beginner to Beyond
  + Що опрацьовано:
    - Визначення та властивості констант та змінних
    - Приклади використання констант та змінних у програмуванні
    - Практичне застосування констант та змінних у програмуванні
  + Статус: Ознайомлений

# **Виконання роботи:**

## **1. Опрацювання завдання та вимог до програм та середовища:**

**Завдання №1 Опрацювання теорії**

* Деталі завдання: опрацювання різноманітних матеріалів, аналіз та дослідження відео, статей, книг на задані теми. Організація та структурування отриманих даних для можливості ефективніше засвоювати отримані знання на практиці. Вивчення найважливіших моментів.

**Завдання №2 Проектування за допомогою Draw.io(складання блок-схем)**

* Деталі завдання:

Побудова блок-схем до завдань 3-9.

**Завдання №3 Лабораторна №1 з VNS, завдання 1**

* Варіант завдання: **6**
* Деталі завдання:

математичні обчислення, використовуючи бібліотеку <cmath>, для того,щоб обчислити задану арефметичну формулу

* Важливі деталі для врахування в імплементації програми

Фонкція степеня pow, а також робота коду зі змінною double і float, проаналізувати чому виходять різні результати.

**Завдання №4 Лабораторна №1 з VNS, завдання 2**

* Варіант завдання: **6**
* Деталі завдання:
* Порівняння чисел і в залежності від чисел, котрі ввів користувач. Вивід в консоль відповідні порівняння або обчислення цих чисел.

Increment and decrement у виконанні програми

* Важливі деталі:

Increment and decrement

Розуміти в чому різниця між ++n та n++

**Завдання №5 Лабораторна №2 з VNS, завдання 1**

* Варіант завдання: **20**
* Деталі завдання: Порахувати суму ряду

**Завдання №6 Лабораторна №3 Algotester, завдання 1**

* Варіант завдання: **1**
* Деталі завдання: Є 5 кубів зі сторонами a1…5. Користувач будує піраміду, ставлячи кожен наступний куб на попередній. Якщо об’єм наступного куба більший за об’єм куба на вершині піраміди, ви програєте. Завдання: визначити, як закінчиться гра

**Завдання №7 Лабораторна №1 Algotester, завдання 2**

* Варіант завдання: **2**
* Деталі завдання: Є дорога, представлена як масив чисел. Завдання: знайти мінімальну можливу втому після видалення одного числа.
* Важливі деталі:

Втому визначає різниця між максимальним та мінімальним елементом.

Можна видалити одне число з масиву, щоб мінімізувати втому.

**Завдання №8 Виконання практичної роботи №2**

* Деталі завдання: створити простий порадник щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди.

Вимоги: запропонувати користувачеві ввести поточні погодні умови(sunny, rainy, cloudy, snowy,windy)

* Важливі деталі:

**Завдання №9 Виконання самостійної роботи**

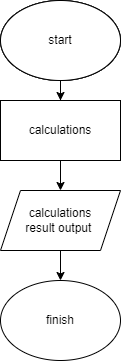
* Деталі завдання:

Виконання завдавння №7 за допомогою іншого алгоритма (merge sort)

## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма № 1 VNS Lab 1 – Task 1-6

* Блок-схема

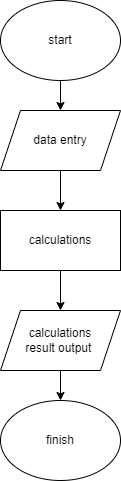


* Планований час на реалізацію: 5хв

використання бібліотеки cmath

Програма №2 VNS Lab 1 – Task 2-6

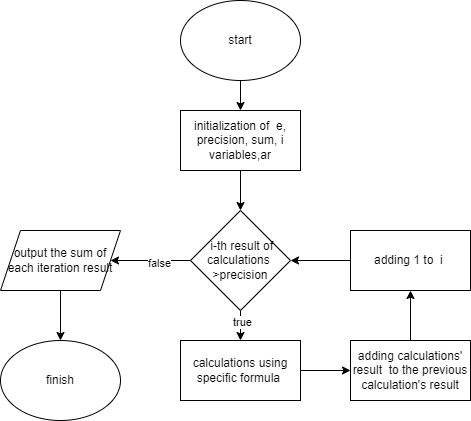
* Блок-схема



* Планований час на реалізацію: 5хв

Програма №3 VNS Lab 2-20

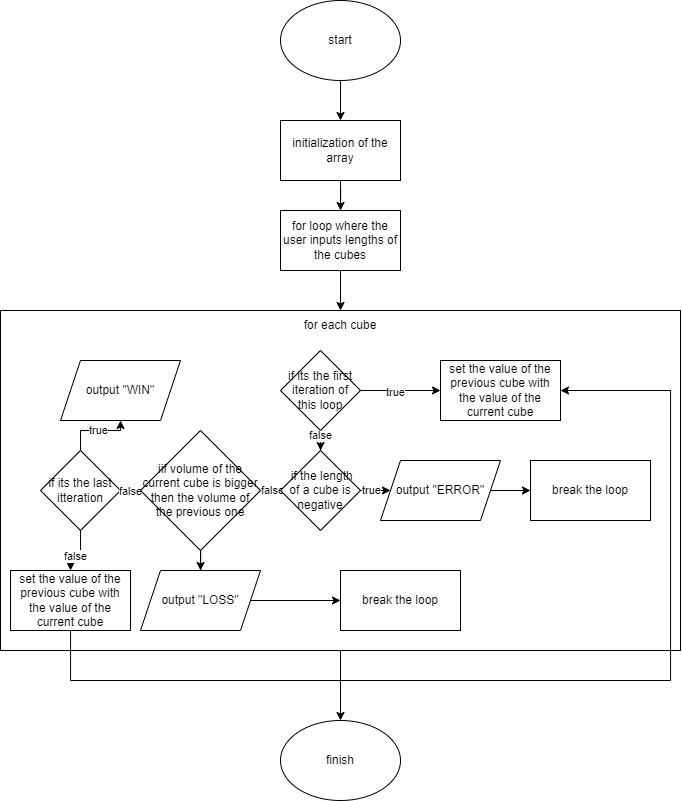
* Блок-схема



* Планований час на реалізацію 15хв

Програма №4 Lab 1v3

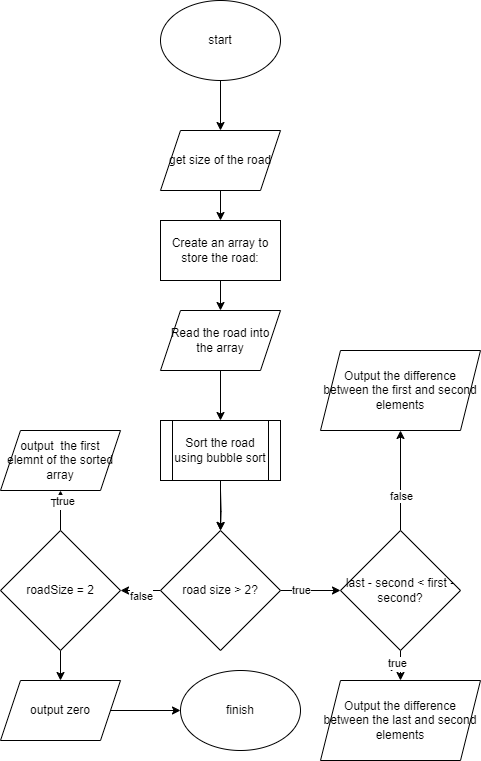
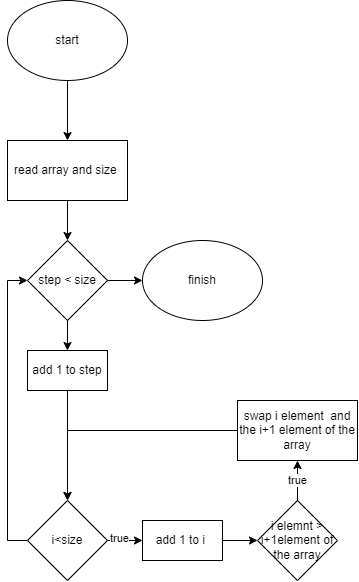
* Блок-схема



* Планований час на реалізацію 30 хв

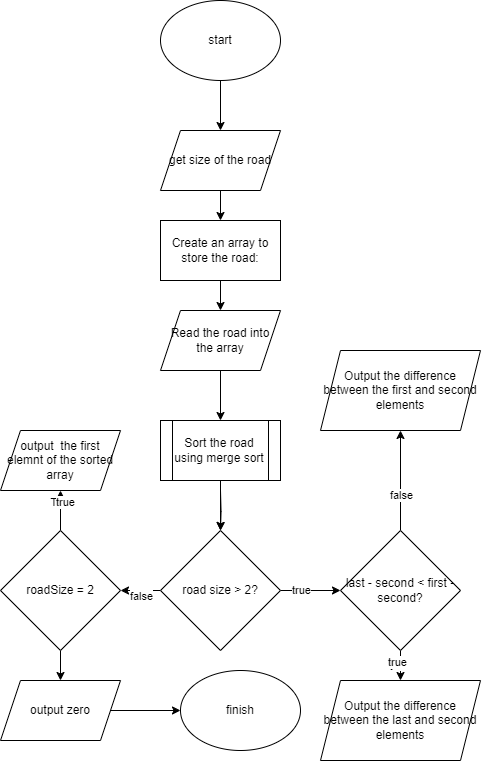
Програма №5 Lab 2v1-bbl

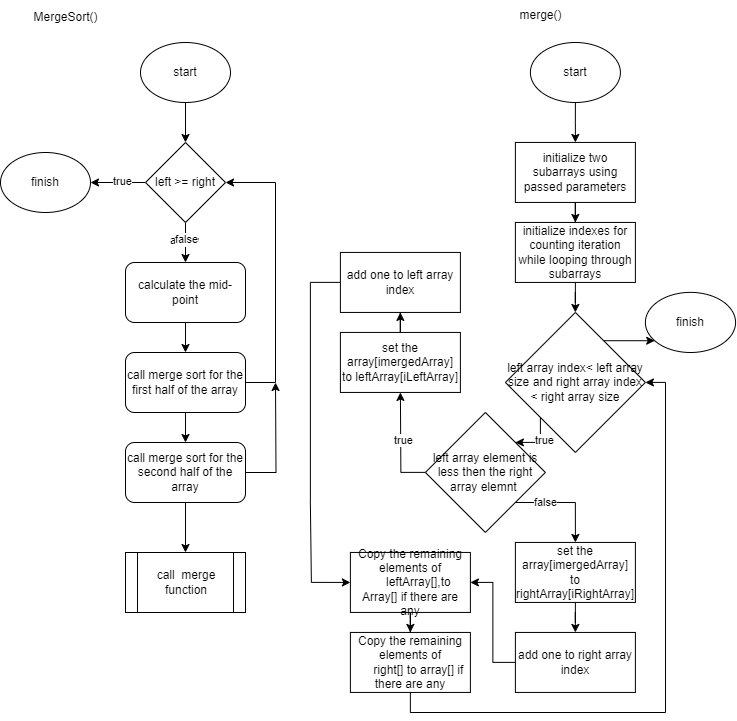
* Блок-схема

* Планований час на реалізацію: 50хв

Програма №6 Lab 2v1-slct- self –practice

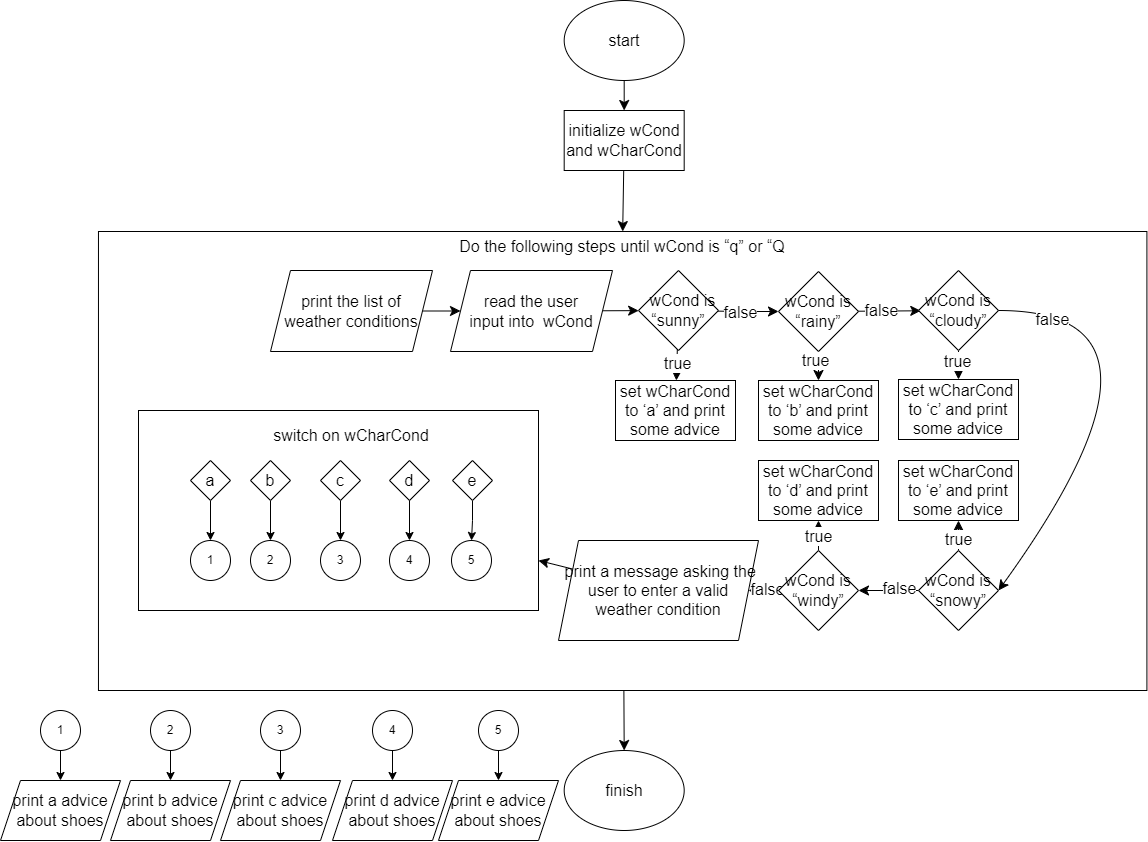
* Блок-схема
* 



* Планований час на реалізацію:1год

Програма №7 Class Practice Task

* Блок-схема



* Планований час на реалізацію 15хв

## **4. Код програм з посиланням на зовнішні ресурси:**

#include <cmath>

#include <iostream>

using namespace std;

int main()

{

    int a {1000};

    double b {0.0001};

    float r {0.0001};

    double c = (pow((a-b),3) - (pow(a,3) - 3\*a\*pow(b,2)))/(pow(b,3)-3\*pow(a,2)\*b);

    float k = (pow((a-r),3) - (pow(a,3) - 3\*a\*pow(r,2)))/(pow(r,3)-3\*pow(a,2)\*r);

   cout << c << endl;

   cout << k << endl;

  return 0 ;

}

VNS Lab 1 – Task 1-6 та №1 [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-097297c805c8250774272aa6616b4bbbdd5bd20fde5f2b0cf8b246d98169163d](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-097297c805c8250774272aa6616b4bbbdd5bd20fde5f2b0cf8b246d98169163d)

#include <iostream>

using namespace std;

int main()

{

    int m {0};

    int n {0};

    cin >> m;

    cin >> n;

    cout << m-++n << endl;

    bool first = ++m > --n;

    bool second = --n < ++m;

    cout <<  first << endl;

    cout <<  second << endl;

    return 0;

}

VNS Lab 1 – Task 2-6 та №2 [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-c65ff5ba37e282d2bd2dfcbec43afc7d9fa6ae2da5ff0ecd3a008de8272e0d51](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-c65ff5ba37e282d2bd2dfcbec43afc7d9fa6ae2da5ff0ecd3a008de8272e0d51)

#include <math.h>

#include <iostream>

using namespace std;

int main()

{

    const double e {2.71828};

    double precision {0.0001};

    double sum {0};

    int i {1};

    double ar {pow(e,i)\*pow(100, -pow(i, 2))};

    while (abs(ar)  > precision)

    {

        ar = pow(e,i)\*pow(100, -pow(i, 2));

        sum += ar;

        i++;

    }

    cout << sum <<endl;

    return 0;

}

VNS Lab 2-20 та №3 [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-6e3e95e0b9d891c35730f1241a57d9fb20eacfd9133e0e691b7c0f83c864a62b](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-6e3e95e0b9d891c35730f1241a57d9fb20eacfd9133e0e691b7c0f83c864a62b)

#include <iostream>

#include <math.h>

using namespace std;

int main()

{

    long long int an[5] {0};

    long long int prevCubeV {0};

   for (size\_t i = 0; i < 5; i++)

   {

    // cout << "enter the length of the "<<i+1<<"  cube:"<<endl;

    cin>> an[i];

    if(an[i] <= 0 || an[i] <= -pow(10, 12) || an[i] >= pow(10, 12)){

            cout << "ERROR"<< endl;

            return 0;

        }

    long long int cubeV = pow(an[i],3);

        // cout << i+1 << " cube:" << cubeV<< endl;

        if (i == 0) {

            prevCubeV = cubeV;

        }

        else{

            // cout << "comparing " << i << " with "<< i+1<< " ..."<< endl;

            if(an[i-1] < an[i]&& i !=0){

                cout << "LOSS"<<endl;

                return 0;

            }

            else if (i == 4){

                cout << "WIN" <<endl;

                return 0;

            }

            prevCubeV = cubeV;

        }

   }

    return 0;

}

Lab 1v3 та №4 [EPIC-2 Dmytro Yunko code of solved problems added by tondeee · Pull Request #166 · artificial-intelligence-department/ai\_programming\_playground (github.com)](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/166/files)

#include <iostream>

using namespace std;

void swap(int \*a, int \*b) {

  int temp = \*a;

  \*a = \*b;

  \*b = temp;

}

void sort(int array[], int size) {

   for (int step = 0; step < size - 1; step++) {

    for (int i = 0; i < size - 1; i++) {

      if (array[i] > array[i+1])

      swap(&array[i], &array[i+1]);

    }

  }

}

int main()

{

    int roadSize {0};

    cin >> roadSize;

    int road[roadSize] {0};

    int min {0};

    for (size\_t i = 0; i < roadSize; i++)

    {

        cin >> road[i];

    }

    sort(road, roadSize);

    if(roadSize > 2){

      if(road[roadSize-1] - road[1] < road[roadSize-2] - road[0]){

        cout << road[roadSize-1] - road[1] <<endl;

      }else{

        cout << road[roadSize-2] - road[0] <<endl;

      }

    for (size\_t i = 0; i < roadSize; i++)

    {

      cout << road[i];

    }

    }else if (roadSize == 2 ){

      cout << road[0];

    }else{

      cout << 0;

    }

    return 0;

}

Lab 2v1-bbl та №5 [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-fa496ce64214e02558a182a77687c962afccf39b5e993c8fafec99af9053745a](%20https:/github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-fa496ce64214e02558a182a77687c962afccf39b5e993c8fafec99af9053745a)

#include <iostream>

using namespace std;

void merge(int array[],int left,int right, int mid ) {

  int leftArraySize = mid - left + 1;

  int rightArraySize = right - mid;

  int leftArray [leftArraySize] {0};

  int rightArray [rightArraySize] {0};

 for (size\_t i = 0; i < leftArraySize; i++){

      leftArray[i] = array[left + i];

    }

    for (size\_t j = 0; j < rightArraySize; j++){

        rightArray[j] = array[mid + 1 + j];

    }

    int iLeftArray = 0;

    int iRightArray = 0;

    int imergedArray = left;

    while (iLeftArray < leftArraySize && iRightArray < rightArraySize){

      if (leftArray[iLeftArray]<= rightArray[iRightArray]) {

            array[imergedArray] = leftArray[iLeftArray];

            iLeftArray++;

        }

        else {

            array[imergedArray]= rightArray[iRightArray];

            iRightArray++;

        }

        imergedArray++;

    }

    while (iLeftArray < leftArraySize) {

        array[imergedArray]= leftArray[iLeftArray];

        iLeftArray++;

        imergedArray++;

    }

    while (iRightArray < rightArraySize) {

        array[imergedArray]= rightArray[iRightArray];

        iRightArray++;

        imergedArray++;

    }

}

void mergeSort(int array[], int left, int right)

{

    if (left >= right)

        return;

    int mid = left + (right - left) / 2;

    mergeSort(array, left, mid);

    mergeSort(array, mid + 1, right);

    merge(array, left, right, mid);

}

int main()

{

    int roadSize {0};

    cin >> roadSize;

    int road[roadSize] {0};

    int min {0};

    for (size\_t i = 0; i < roadSize; i++)

    {

        cin >> road[i];

    }

    mergeSort(road, 0, roadSize-1);

    if(roadSize > 2){

      if(road[roadSize-1] - road[1] < road[roadSize-2] - road[0]){

        cout << road[roadSize-1] - road[1] <<endl;

      }else{

        cout << road[roadSize-2] - road[0] <<endl;

      }

    }else if (roadSize == 2 ){

      cout << road[0];

    }else{

      cout << 0;

    }

    return 0;

}

Lab 2v1-merge sort – self -practice та №6 [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-7c4ae139d08873378ddec2baaec9e4b713feac709e047c5f66da6966feae92d8](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-7c4ae139d08873378ddec2baaec9e4b713feac709e047c5f66da6966feae92d8)

#include <iostream>

#include <string>

using namespace std;

int main()

{

    string wCond ;

    char wCharCond;

    do

    {

         cout <<"\n" << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<< endl << "\n";

        cout << " sunny;" << endl;

        cout << " rainy" << endl;

        cout << " cloudy" << endl;

        cout << " snowy;" << endl;

        cout << " windy;" << endl;

        cout << "Q = quite" << endl;

        cout << "\nenter your weather conditions:";

        cin >> wCond;

        if(wCond == "sunny"){

            wCharCond= 'a';

            cout << "you arent gonna need any jacket, dont worry)"<< endl;

            cout << "what a great day for a picnic BTW"<< endl;

            // cout << "Take your favourite shoes"<< endl;

        }else if(wCond == "rainy"){

            wCharCond= 'b';

            cout << "You should take a jacket with you"<< endl;

            cout << "The best weather for a reading indoors "<< endl;

            // cout << "rubber boots - a great idea!"<< endl;

        }else if(wCond == "cloudy"){

             wCharCond= 'c';

            cout << "You should take a jacket with you just in case"<< endl;

            cout << "How about visiting a museum today?"<< endl;

            // cout << "You can wear anything you want today"<< endl;

        }else if(wCond == "snowy"){

             wCharCond= 'd';

            cout << "You must take a jacket with you"<< endl;

            cout << "A great time to make a snowman"<< endl;

            // cout << "snow boots will warm your feet!"<< endl;

        }else if(wCond == "windy"){

             wCharCond= 'e';

            cout << "You should take a jacket with you"<< endl;

            cout << "Try to lunch a kite if you have one "<<endl;

            // cout << "take some sturdy shoes "<<endl;

        }else{

            cout << "Try to enter the weather condition from the list above"<< endl;

        }

        switch (wCharCond)

        {

        case 'a':{

            cout << "Take your favourite shoes"<< endl;

            break;

        }

        case 'b':{

             cout << "rubber boots - a great idea!"<< endl;

            break;

        }

        case 'c':{

             cout << "You can wear anything you want today"<< endl;

            break;

        }

        case 'd':{

            cout << "snow boots will warm your feet!"<< endl;

            break;

        }

        case 'e':{

            cout << "take some sturdy shoes "<<endl;

            break;

        }

        default:

            break;

        }

    } while (wCond != "q" && wCond !="Q" );

    return 0;

}

## Class Practice Task [https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/230/files#diff-bfc0f8dc22431c483e2b9f70ae259ca25aaeec3f6772fcfdd8c5711bd6a4f6e6](https://github.com/artificial-intelligence-department/ai_programming_playground/pull/230/files%23diff-bfc0f8dc22431c483e2b9f70ae259ca25aaeec3f6772fcfdd8c5711bd6a4f6e6)

## **5.Результати виконання завдань, тестування та фактично затрачений час:**

Завдання №3 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\lab\_01\output> & .\'task\_01.exe'

1.2207

PS D:\ED\University\C++\epic\_02\lab\_01\output> & .\'task\_01.exe'

1

VNS Lab 1 – Task 1-6 та №1 до блоку з виконанням та тестуванням програми

Завдання №4 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\lab\_01\output> & .\'task\_02.exe'

0 0

-1

1

1

VNS Lab 1 – Task 2-6 та №2 до блоку з виконанням та тестуванням програми

Завдання №5 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\lab\_02\output> & .\'lab\_02k.exe'

0.0271829

VNS Lab 2-20 та №3 до блоку з виконанням та тестуванням програми

Завдання №6 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\lab\_02\output> & .\'lab\_02k.exe'

0.0271829

PS D:\ED\University\C++\epic\_02\alg\_lab\_01\output> & .\'lab.exe'

enter the length of the 1 cube:

10

enter the length of the 2 cube:

9

enter the length of the 3 cube:

8

enter the length of the 4 cube:

7

enter the length of the 5 cube:

6

WIN

PS D:\ED\University\C++\epic\_02\alg\_lab\_01\output> & .\'lab.exe'

enter the length of the 1 cube:

4

enter the length of the 2 cube:

3

enter the length of the 3 cube:

2

enter the length of the 4 cube:

1

enter the length of the 5 cube:

0

ERROR

PS D:\ED\University\C++\epic\_02\alg\_lab\_01\output> & .\'lab.exe'

enter the length of the 1 cube:

1

enter the length of the 2 cube:

1

enter the length of the 3 cube:

1

enter the length of the 4 cube:

1

enter the length of the 5 cube:

2

LOSS

Lab 1v3 та №4 до блоку з виконанням та тестуванням програми

Завдання №7 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 2 2 4 4

2

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 2 2 2 4

1

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 3 3 4 4

1

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

1

5

0

Lab 2v1-bbl та №5 до блоку з виконанням та тестуванням програми

Завдання №8 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 2 2 4 4

2

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 2 2 2 4

1

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

5

1 3 3 4 4

1

PS D:\ED\University\C++\epic\_02\alg\_lab\_02\output> & .\'alg\_lab\_02\_bbl.exe'

1

5

0

Lab 2v1-merge sort – self -practice та №6 до блоку з виконанням та тестуванням програми

Завдання №9 Деталі по виконанню і тестуванню програми

PS D:\ED\University\C++\epic\_02\prac1\output> & .\'prac1.exe'

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sunny;

rainy

cloudy

snowy;

windy;

Q = quite

enter your weather conditions:sunny

you arent gonna need any jacket, dont worry)

what a great day for a picnic BTW

Take your favourite shoes

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sunny;

rainy

cloudy

snowy;

windy;

Q = quite

enter your weather conditions:rainy

You should take a jacket with you

The best weather for a reading indoors

rubber boots - a great idea!

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sunny;

rainy

cloudy

snowy;

windy;

Q = quite

enter your weather conditions:cludy

Try to enter the weather condition from the list above

rubber boots - a great idea!

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sunny;

rainy

cloudy

snowy;

windy;

Q = quite

enter your weather conditions:snowy

You must take a jacket with you

A great time to make a snowman

snow boots will warm your feet!

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

sunny;

rainy

cloudy

snowy;

windy;

Q = quite

enter your weather conditions:windy

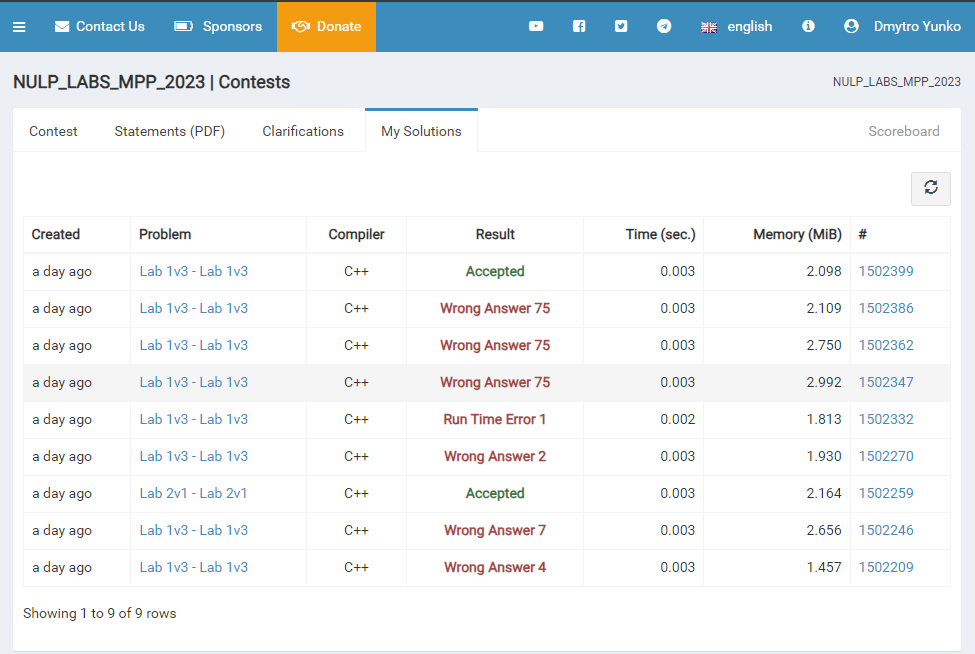
You should take a jacket with you

Try to lunch a kite if you have one

take some sturdy shoes

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Class Practice Task №7 до блоку з виконанням та тестуванням програми



Algotester report

Час затрачений на виконання завдання: 3дн

# **Висновки:**

В процесі вивчення тем “Лінійні алгоритми та розгалужені алгоритми”, “Умовні оператори” та “Константи та змінні” було отримано глибокі знання та практичні навички. Це допомогло зрозуміти основи програмування на мовах C/C++ і набути впевненості в написанні програм.

Вивчення лінійних та розгалужених алгоритмів допомогло зрозуміти, як можна структурувати програму для вирішення різних задач. Умовні оператори дали змогу керувати потоком виконання програми, а константи та змінні стали основою для збереження та маніпулювання даними.

Використання цих знань і навичок у подальшому дозволить ефективно розв’язувати задачі програмування, покращити якість коду і сприятиме продуктивності роботи. Завдяки цьому курсу, я почуваю себе більш підготовленим до подальшого вивчення програмування і з нетерпінням чекаю нових викликів.

Цей звіт показав, що систематичне вивчення теоретичного матеріалу, поступове опрацювання кожної теми і практичне застосування отриманих знань є ключем до успішного опанування програмування. Завдяки цьому досвіду, я маю всю необхідну базу для подальшого вивчення мов програмування C++ і розумію, як ефективно застосовувати ці навички на практиці.